Use of Controlled Vocabularies in USGS Information Applications: Requirements Analysis for Automated Processes and Services (Community for Data Integration Project CDI14-LF336)

Use Case Documentation (Revised)

What are use cases?

Use cases represent the first step in the orderly development of a new computer system or business process. Use cases focus on the interaction between external *actors* and the *system under development* (SuD), revealing how actor *goals* are fulfilled by system *behavior*. A single SuD may require several use cases to describe its interaction with all of the identified actors (human and otherwise). System behavior as revealed by these use cases translates into formal system requirements, thus informing the rest of the design process—such as the selection of a particular technological approach.

A use case is often preceded by a *user story*, which describes actor goals and system behavior informally in a few sentences. The use case that follows is a more structured narrative, including actor descriptions and a step-by-step *basic flow* of events, along with *alternate flows* as necessary to describe unusual or contingent system behavior. Although use cases are primarily textual, they are usually supported by graphical elements such as activity diagrams and conceptual models.

Use cases are best developed by small teams with varied skills and viewpoints. Ideally, the use case team will include a facilitator, note taker, domain experts, conceptual modelers, and software engineers. The team develops the use cases and distills the desired SuD behavior into formal requirements that are passed to the system designers—but first, an independent review panel provides a fresh set of eyes to insure that no important system behavior has been overlooked. Review, feedback, and revision may occur repeatedly: use case development is fundamentally an *iterative* process.

Developing use cases is easier said than done, however, as many authorities on the subject have noted. Bittner and Spence (2003, p. 213), for instance, cautioned against use case teams focusing on internal design considerations rather than system behavior from the actor's perspective:

> The whole point of the use case is to capture a description of something that the system must do. It is an expression of a desired behavior of the system. The system must behave that way no matter how it is designed and implemented. Its value is in expressing that behavior in a simple and unambiguous way. … This is easy to say, but hard to follow. Software developers often seem to be unable to help themselves: They begin to talk of "levels" of use cases, and soon enough the decomposition begins. … Pretty soon, the model looks a lot like a high-level design of the system and not at all like a description of what the system is supposed to do from an external observer's perspective.

Bittner and Spence have made two important points: first, use cases provide a fundamentally *synthetic* view of actor-system interaction, whereas the functional decomposition procedures familiar to software developers are fundamentally *analytic*; second, the use case team treats the SuD as a *black box*, leaving the design considerations for later.

Another practitioner, Susan Lilly (1999), has compiled a list of common pitfalls in use case development, including ambiguous system boundaries, scoping problems, and so on. As for the presentation of use cases, Cockburn (2001) has provided the most basic guiding principle: "Write something readable. *Casual, readable use cases are still useful, whereas unreadable use cases won't get read.*"

With all of this good advice, we've found that it's still easy to go wrong—but these lessons are best learned through trial and error, employing an iterative process based on review, feedback, and revision.

Project Use Cases

The team developed three use cases for the project, following the methodology of Fox and McGuinness (2008):

- Use Case 1, "Assign keywords to a metadata record using one or more controlled vocabularies," develops the functional requirements for a vocabulary server interacting with a metadata creation tool.
- Use Case 2, "A catalog search interface uses vocabulary services to help users find data," develops functional requirements for the same vocabulary server, but this time interacting with a catalog user interface.
- Use Case 3, "Create specialized indexes to enhance the searchability of metadata," develops the requirements if the vocabulary server is used by a catalog system to develop an internal table that cleans up and cross-references keywords that are found in metadata records

The text of each use case was developed using a standard template, which includes sections for summarizing the goal of the use case, outlining the basic flow of events necessary to achieve that goal (along with alternate flows as necessary), and noting supporting information. In addition to this narrative, each use case includes an activity diagram illustrating the basic flow, along with conceptual models that serve as a "reality check" for the system designers by clarifying the entities and relationships within the conceptual domain of the prototype server.

Analysis of the three use cases produced the preliminary set of system requirements shown in table 1. Most of these requirements are indicated by two or even three of the use cases, giving us confidence that our analysis provides a solid foundation for designing the prototype server and its services.

The use case documents that were provided to the review panel are available (along with the review panel's report) on the CDI confluence site at <https://my.usgs.gov/confluence/display/cdi/Use+Cases+for+Vocabulary+Web+Services>. Revised use case documents, taking into account feedback from the review panel, are shown below.

Table 1: Functional requirements discovered through analysis of use cases. “CVS” refers to a specific controlled vocabulary set that is provided by the vocabulary server.

Service Description	**Case 1**	**Case 2**	**Case 3**
Server provides list of all available CVS identifiers.	x		x
Server provides list of all available CVS names, both preferred and alternative.	x		x
Given a CVS identifier, server provides a description of a single CVS (producer, version, identifier, preferred name, recommended use, etc.)	x		x
Given a CVS identifier, server indicates whether broader and narrower relationships in that CVS are transitive.	x		x
Given a CVS identifier, server indicates whether that CVS has a hierarchical structure.	x		x
Given a CVS identifier, server provides a list of top level terms or recommended starting terms for browsing that CVS.		x	
Given a CVS identifier and a choice of hierarchical level, server provides a list of all terms included on that level in a single CVS.	x		
Given a search string (possibly including wildcards) and a CVS identifier, server provides terms that match the given string within the CVS.	x	x	x
Given a search string (possibly including wildcards), server provides terms that match the given string within all available CVSes.	x	x	x
Given a search term and a CVS identifier, server provides a description of the given term as specified in the CVS (identifier, scope notes, etc.).	x	x	
Given a search term and a CVS identifier, server provides the set of non-preferred terms listed in the CVS for the concept identified by the given term.		x	
Given a search term and a CVS identifier, server provides the set of other terms within the CVS that are related to the given term.	x	x	
Given a search term and a CVS identifier, server provides the set of other terms within the CVS that are broader than the given term.	x	x	x
Given a search term and a CVS identifier, server provides the set of other terms within the CVS that are narrower than the given term.	x	x	

Use Case 1 (Revised Narrative)

Use Case Name: Assign keywords to a metadata record using one or more controlled vocabularies
Point of Contact Name: Drew Ignizio

Use Case Name

Give a short descriptive name for the use case to serve as a unique identifier. Consider goal-driven use case name.

Assign keywords to a metadata record using one or more controlled vocabularies

Goal

The goal briefly describes what the user intends to achieve with this use case.

To assign keywords to a metadata record with the aid of a metadata creation tool and services provided by a controlled-vocabulary server.

Summary

Give a summary of the use case to capture the essence of the use case (no longer than a page). It provides a quick overview and includes the goal and principal actor.

A metadata author or editor (“user”) wishes to select controlled-vocabulary terms that add value to a metadata record, using a metadata creation tool such as the Metadata Wizard. In this use case the metadata tool acts as an intermediary between the user and services provided by a controlled-vocabulary server.

The metadata tool presents the user with the names and characteristics of vocabularies available from the vocabulary server. The user selects one or more relevant vocabularies from which to draw keywords. For each vocabulary, in turn, the user either searches for a term or browses a list of available terms to select keywords that describe a given data set. The metadata tool writes the selected terms to the metadata record and also identifies the vocabularies from which they came.

The metadata tool might also provide information about the context and meaning of individual vocabulary terms to assist the user in determining whether the terms are appropriate for a given metadata record. The tool might also use vocabulary characteristics to determine which vocabularies are offered for specific fields (place keywords, theme keywords, etc.) within a metadata record.

Ideally, the interaction between the metadata tool and the vocabulary server will take place quickly enough to provide a relatively uninterrupted user experience.

Actors and SuD

List actors, people or things outside the system that either acts on the system (primary actors) or is acted on by the system (secondary actors). Primary actors are ones that invoke the use case and benefit from the result. Identify sensors, models, portals and relevant data resources. Identify the primary actor and briefly describe role.

Metadata author or editor – Actor A (secondary actor, a human)

Metadata tool – Actor B (primary actor, a machine)

Vocabulary services provided by a vocabulary server (not an actor, rather the system under development, or SuD)

Preconditions

Here we state any assumptions about the state of the system that must be met for the trigger (below) to initiate the use case. Any assumptions about other systems can also be stated here, for example, weather conditions. List all preconditions.

- The metadata tool has access via Internet to the vocabulary services provided by the vocabulary server (the SuD).
- Controlled vocabularies are available through the vocabulary services in a form that the metadata tool can use.
- The metadata tool is available to the metadata author or editor.
- The benefit of using controlled vocabularies to improve metadata records for discovery and use is accepted by the metadata author or editor and by the custodians of downstream systems that harvest and organize metadata in searchable collections.
- Controlled vocabularies provided will contain correct and quality-controlled values.

Triggers

Here we describe in detail the event or events that brings about the execution of this use case. Triggers can be external, temporal, or internal. They can be single events or when a set of conditions are met, List all triggers and relationships.

The metadata tool begins execution of the use case when a metadata author or editor reaches the keyword stage in the metadata creation process.

Basic Flow

Often referred to as the primary scenario or course of events. In the basic flow we describe the flow that would be followed if the use case where to follow its main plot from start to end. Error states or alternate states that might be highlighted are not included here. This gives any browser of the document a quick view of how the system will work. Here the flow can be documented as a list, a conversation or as a story.(as much as required)

1) Actor B requests a list of vocabularies available from the vocabulary server, which the service returns with additional information about each vocabulary, such as:

- Server-specific identifier for the vocabulary.
- The specific version of the vocabulary provided by the server.
- A description of the purpose and extent of the vocabulary, including the sorts of keywords that the vocabulary provides (in CSDGM metadata: theme keywords? place keywords? stratum keywords? temporal keywords?).
- The level of granularity in the vocabulary.

- Information about who produced the vocabulary.
- Information about the structure and organization of a vocabulary: for example, number of top-level terms, number of descriptors, flat or hierarchical structure, mono- or multi-lingual terms, whether it conforms strictly to SKOS or extended SKOS, RDF root identifier, suggested RDF prefix, etc.
- A standardized name for the vocabulary (e.g., the preferred label to be used when listing or referencing the vocabulary in a CSDGM metadata record).
- If available, information on where and how the terms from the vocabulary are being used downstream (i.e., in data clearinghouses or other search catalogs).

2) Actor B presents Actor A with recommended vocabularies (i.e., a subset of the vocabularies provided by the server, based on Actor B's rules). The vocabularies might have some description based on information that was passed back from the vocabulary server, including which vocabularies are appropriate for browsing, etc.

3) Actor A tells Actor B which vocabularies to use.

4) Actor B prompts Actor A to select a search or browse option.

5) If the search option is selected, Actor A provides a search string. If the browse option is selected, Actor A pushes a button that says "Go".

6) Actor B sends a message to the vocabulary server specifying (1) which vocabularies to use, (2) what search string to use, if Actor A has provided one, or (3) if Actor A has chosen a single vocabulary but has not entered a search string, that the server return a default set of terms from the vocabulary to serve as a starting point for browsing. The vocabulary server returns a list of terms relevant to Actor B's request, including contextual information such as scope notes, broader terms, narrower terms, and related terms.

7) Actor B displays the list of terms, possibly including contextual information, and asks Actor A to choose individual terms from the list or to continue exploring the vocabularies.

8) Actor A selects terms from the list (step 8A in activity diagram) or provides a new search string (step 8B in activity diagram). If a new string is provided, the flow continues with step 6.

9) Actor B writes the selected terms and associated vocabulary identification to the metadata record.

10) Actor B asks Actor A if more terms are needed, and repeats to step 3 if so.

Alternate Flow

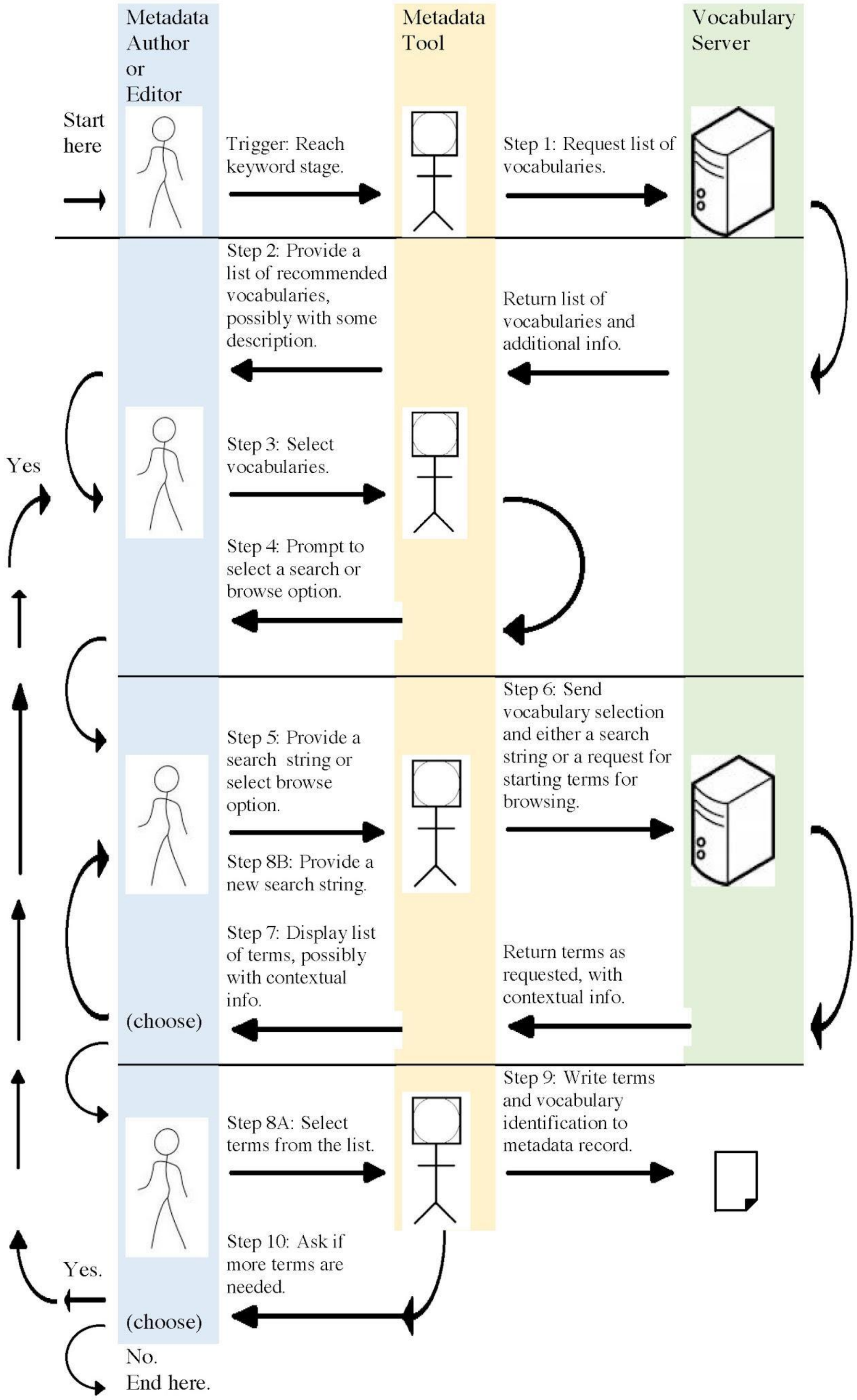
Here we give any alternate flows that might occur. May include flows that involve error conditions. Or flows that fall outside of the basic flow.

1) If vocabulary services are down, the use case ends.

2-4) Steps 2-4 could be reordered, allowing Actor A to enter a search string without selecting vocabularies first.

8) If Actor A gives up on finding an appropriate term, the use case ends.
Post Conditions *Here we give any conditions that will be true of the state of the system after the use case has been completed.* The metadata record that is produced includes keywords from one or more controlled vocabularies, with each keyword attributed to its source vocabulary (the “keyword thesaurus” in CSDGM metadata).
Activity Diagram *Here a diagram is given to show the flow of events that surrounds the use case. It might be that text is a more useful way of describing the use case. However often a picture speaks a 1000 words.* See following page.

Use Case 1: Assign keywords to a metadata record using one or more controlled vocabularies



Trigger: Reach keyword stage.

Step 1: Request list of vocabularies.

Step 2: Provide a list of recommended vocabularies, possibly with some description.

Return list of vocabularies and additional info.

Step 3: Select vocabularies.

Step 4: Prompt to select a search or browse option.

Step 5: Provide a search string or select browse option.

Step 6: Send vocabulary selection and either a search string or a request for starting terms for browsing.

Step 8B: Provide a new search string.

Step 7: Display list of terms, possibly with contextual info.

Return terms as requested, with contextual info.

Step 8A: Select terms from the list.

Step 9: Write terms and vocabulary identification to metadata record.

Step 10: Ask if more terms are needed.

Notes

There is always some piece of information that is required that has no other place to go. This is the place for that information.

There will be requirements that vocabularies must meet to be compatible with these services and tools. Among these requirements are identifiers for the vocabularies and some general characteristics such as specificity, breadth, range of topics, and language.

Data producers are required to produce metadata per USGS and other federal mandates.

The USGS Science Data Catalog (and similar applications) will use keywords to index results; properly using keywords will help USGS science products be found and used more effectively.

Possible motivations:

- Author of metadata cares about making metadata discoverable in new systems that are harvesting metadata, and is convinced that a reasonable number of controlled vocabulary keywords will be effective.
- Author is instructed to include controlled-vocabulary terms in metadata record.

It may also be worth considering preferred methods or best practices for the format used in the service responses (JSON, XML, etc.). Allowing users to structure their requests to specify the preferred format may be valuable as well.

For example of vocabulary services, see: http://www.itis.gov/ws_description.html

Resources

In order to support the capabilities described in this Use Case, a set of resources must be available and/or configured. These resources include data and services, and the systems that offer them. This section will call out examples of these resources.

Data

Data	Type	Characteristics	Description	Owner	Source System
USGS Thesaurus				USGS	<http://www.usgs.gov/science/about/>
Biocomplexity Thesaurus				USGS	<http://www.usgs.gov/core_science_systems/csas/biocomplexity_thesaurus/>

Application Services

Application	Owner	Description	Source System
Metadata Wizard	USGS Fort Collins Science Center		<https://www.sciencebase.gov/catalog/item/50ed7aa4e4b0438b00db080a>

Use Case 2 (Revised Narrative)

Use Case Name: A catalog search interface uses vocabulary services to help users find data

Point of Contact Name: Lisa Zolly

Use Case Name

Give a short descriptive name for the use case to serve as a unique identifier. Consider goal-driven use case name.

A catalog search interface uses vocabulary services to help users find data

Goal

The goal briefly describes what the user intends to achieve with this use case

To make use of vocabulary services and controlled vocabulary terms in the keyword fields of metadata records to improve precision and recall of data catalog searches (see Other Resources section for definition of “precision” and “recall”).

Summary

Give a summary of the use case to capture the essence of the use case (no longer than a page). It provides a quick overview and includes the goal and principal actor

Customers may have trouble locating information in a USGS data catalog because they are unfamiliar with technical terms in particular fields, the terminology used by USGS to describe its data, or the USGS organizational structure. The customers are diverse; some will use specific scientific terms or the names of particular instruments, while others will use plain language.

The data catalog could use vocabulary services to suggest additional terms that the user might want to search—for example, synonyms, narrower terms, broader terms, related terms, or more technical terms that correspond to the search text the user has entered or chosen.

Part of the catalog search interface might use the information provided by the vocabulary services to explain what the terms mean: scope notes describe how terms are applied, while related terms, narrower terms, and broader terms provide additional context.

A typical scenario would begin when a catalog user selects a keyword from a list or types a search string in a box. Before executing the search, the catalog search interface passes the search string to the vocabulary services to find exact matches, along with synonyms, variants, broader or narrower terms, and related terms that might help the user refine or expand the original search. These suggestions could be provided in real time (as the catalog user types the search string) or after the initial search has been executed. The basic flow of the use case describes both types of assistance to the user.

Actors and SuD

List actors, people or things outside the system that either acts on the system (primary actors) or is acted on by the system (secondary actors). Primary actors are ones that invoke the use case and benefit from the result. Identify sensors, models, portals and relevant data resources. Identify the primary actor and briefly describe role.

Actor A, a catalog user looking for data (secondary actor, a human).

Actor B, the catalog search interface (primary actor, a machine).

SuD, the vocabulary services provided by a vocabulary server (not an actor, rather the system under development).

Preconditions

Here we state any assumptions about the state of the system that must be met for the trigger (below) to initiate the use case. Any assumptions about other systems can also be stated here, for example, weather conditions. List all preconditions.

The described functionality connects and makes use of controlled vocabulary terms that have been included in metadata records (Use Case 1).

The catalog search interface (Actor B) allows a catalog user (Actor A) to type in search terms or browse ("click") through an initial browse list of keywords.

The catalog search interface (Actor B) has protocols for populating an initial browse list of keywords.

Prior to interaction between the catalog search interface (Actor B) and the vocabulary services (SuD), a default vocabulary in the server has been designated.

Triggers

Here we describe in detail the event or events that brings about the execution of this use case. Triggers can be external, temporal, or internal. They can be single events or when a set of conditions are met, List all triggers and relationships.

The catalog user (Actor A) arrives at the search interface for the data catalog (Actor B) and either types a search string in a box or chooses a term from a list.

Basic Flow

Often referred to as the primary scenario or course of events. In the basic flow we describe the flow that would be followed if the use case where to follow its main plot from start to end. Error states or alternate states that might be highlighted are not included here. This gives any browser of the document a quick view of how the system will work. Here the flow can be documented as a list, a conversation or as a story.(as much as required)

1) Actor B (the catalog search interface) offers a browse list of keywords, in addition to a search box that Actor A (the catalog user) can type in. If Actor A starts typing in the box, Actor B sends a rapid-response request to the SuD (vocabulary services) for a list of controlled-vocabulary terms that match Actor A's partial search string. The SuD generates this list from a default vocabulary in the server and returns it to Actor B, who in turn presents it to Actor A. Actor A can choose one of the terms from this list of matches, continue typing in the box, or choose from the original browse list of keywords.

2) Actor B checks the catalog for matches to the search string that Actor A has either entered or chosen. Actor B also sends a request to the SuD to match the complete search string from Actor A against terms from one or more controlled vocabularies in the server.

3) The SuD returns the possible matching terms from the vocabularies along with the complete context of each term (scope notes, broader terms, narrower terms, related terms), which Actor B could use to provide more information for Actor A.

4) Actor B displays the catalog records that match Actor A's search and also suggests the additional search terms from the controlled vocabularies.

5) Actor A has the option of repeating the process until the list of catalog records returned is satisfactory.

Alternate Flow

Here we give any alternate flows that might occur. May include flows that involve error conditions. Or flows that fall outside of the basic flow.

- Additional search terms could be included in the catalog search by default; alternatively, users could be asked if they would like to also include search results based on related terms or other controlled vocabularies.
- Search options could allow users to specify which controlled vocabulary will be used to filter search results; alternatively, a default controlled vocabulary could be applied in all cases.

Post Conditions

Here we give any conditions that will be true of the state of the system after the use case has been completed.

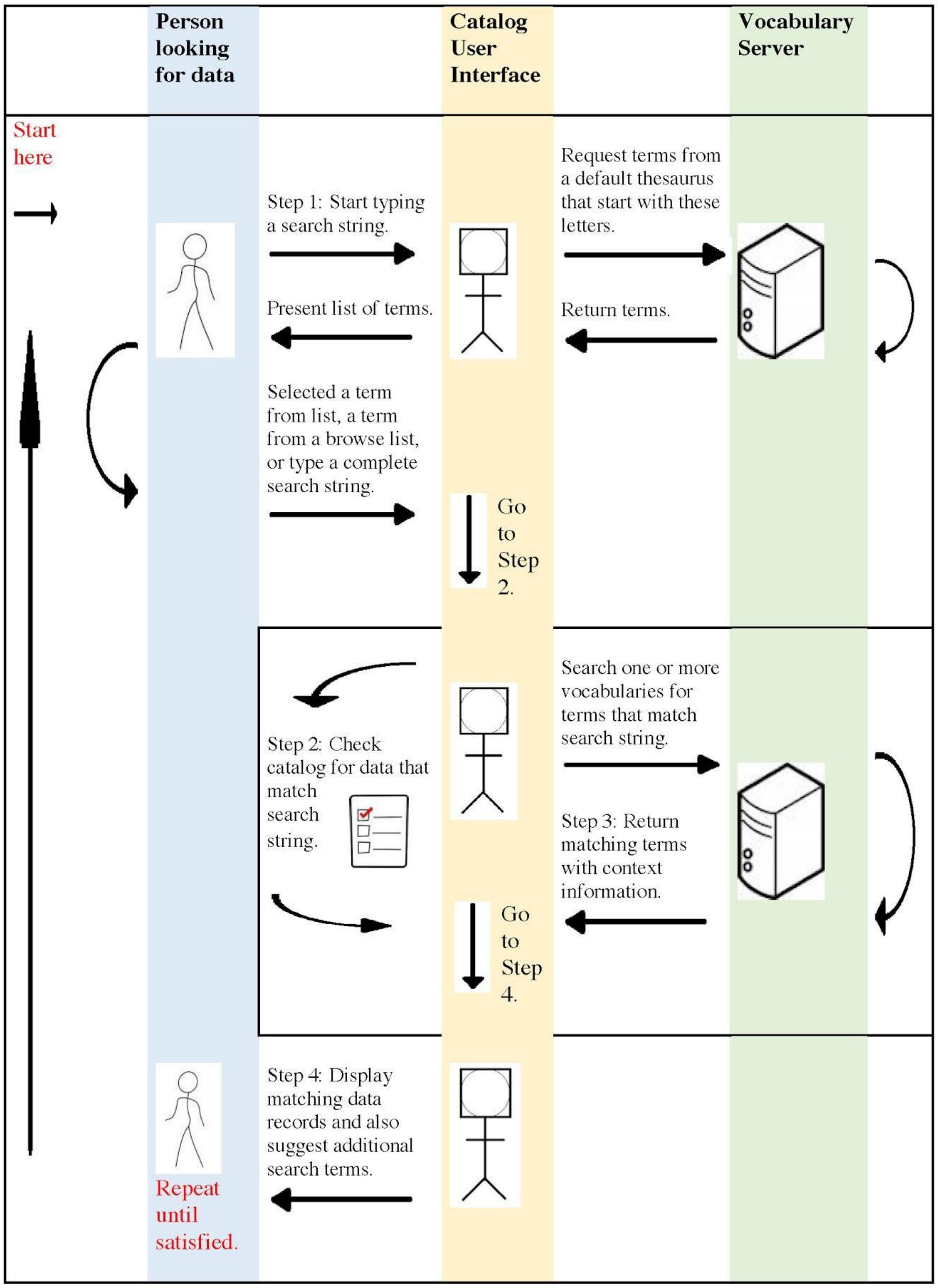
Actor A has a list of data catalog records—perhaps a better list than would have been possible without the use of controlled vocabulary services.

Activity Diagram

Here a diagram is given to show the flow of events that surrounds the use case. It might be that text is a more useful way of describing the use case. However often a picture speaks a 1000 words.

See following page.

Use Case 2: A catalog search interface uses vocabulary services to help users find data



Person looking for data

Catalog User Interface

Vocabulary Server

Start here

Step 1: Start typing a search string.

Request terms from a default thesaurus that start with these letters.

Return terms.

Present list of terms.

Selected a term from list, a term from a browse list, or type a complete search string.

Go to Step 2.

Search one or more vocabularies for terms that match search string.

Step 2: Check catalog for data that match search string.

Step 3: Return matching terms with context information.

Go to Step 4.

Step 4: Display matching data records and also suggest additional search terms.

Repeat until satisfied.

Notes

There is always some piece of information that is required that has no other place to go. This is the place for that information.

If the ThemeKT field in CSDGM metadata contains an identifier of the vocabulary, then the catalog will know which vocabularies are used in the keyword fields of the metadata records.

Future work might include development of crosswalks among vocabularies to help connect plain language and scientific terminology.

Resources

In order to support the capabilities described in this Use Case, a set of resources must be available and/or configured. These resources include data and services, and the systems that offer them. This section will call out examples of these resources.

Other resources

Resource	Owner	Description	Availability	Source System
Definition of precision and recall			<http://en.wikipedia.org/wiki/Precision_and_recall>	Wikipedia

Use Case 3 (Revised Narrative)

Use Case Name: Create a specialized index to enhance the searchability of metadata

Point of Contact Name: Janice Gordon

Use Case Name

Give a short descriptive name for the use case to serve as a unique identifier. Consider goal-driven use case name.

Create a specialized index to enhance the searchability of metadata

Goal

The goal briefly describes what the user intends to achieve with this use case.

To enhance the searchability of metadata by using vocabulary services to help create a specialized index that improves the response of a catalog search interface. This index would provide additional information about the content of metadata records by grouping synonyms, establishing term equivalents between vocabularies, and exploiting hierarchical relationships within vocabularies.

The additional information in the index would be created and linked to the metadata records as they are ingested into the catalog, in order to improve the usability of the records. However, the index would be separate from the original metadata records, which would not be modified.

Summary

Give a summary of the use case to capture the essence of the use case (no longer than a page). It provides a quick overview and includes the goal and principal actor.

A data catalog could improve its response by using the keywords in metadata records to create an index that can be searched more efficiently than the metadata records themselves. The index might simply be an unedited list of the keywords extracted from the metadata records, but an index created in this way would be cumbersome to use if it included singular and plural forms of the same term, variants (including uncorrected misspellings), and too many synonyms. Using vocabulary services to mediate the creation of the index would result in a more useful list of terms, because the keywords supplied by the metadata records could be standardized and augmented to improve the recall of searches (see Other Resources section for definition of “recall”). Some of the ways that a catalog manager could use vocabulary services in this process include the following:

- While ingesting metadata records, the catalog manager could consult vocabulary services to identify controlled vocabulary terms equivalent in meaning to the keywords in the ingested metadata records and then add these vocabulary and term identifiers to the catalog’s internal index. This information would enable additional information about the terms to be fetched from the vocabulary server as needed.

- Multiple controlled vocabulary terms could be provided for each keyword in the metadata record, so that the record would match multiple search texts.
- The controlled vocabulary terms added to the index could include broader terms if they are provided by a thesaurus, a process called “up-posting.” For example, in the USGS thesaurus, the term “mine drainage” is a type of industrial pollution, so every record that was assigned the keyword “mine drainage” would be indexed in such a way that it would be returned in a search for “industrial pollution.”
- Misspellings and other errors could be corrected.
- Weighting of keywords could be employed (e.g., non-controlled terms would be weighted lower than controlled terms).
- Category mistakes could be corrected, in which (for example) valid place keywords have been misused as theme keywords: if “Gulf of Mexico” is ingested as a theme keyword and can be matched to a term in a dictionary of place names (gazetteer), the index could record it as a place keyword. (This is the semantic validation of metadata content.)
- Keywords attributed to a particular keyword thesaurus could be checked to see if they are actually in the referenced vocabulary.
- “Homeless to homes” updates would be possible, in which non-controlled keywords assigned to keyword thesaurus “none” or “general” (in CSDGM metadata) are re-assigned to appropriate controlled vocabularies if a match can be made.

Although the catalog index would be separate from the metadata records themselves, the process of creating the index might identify ways in which the original metadata records could be improved (for instance, by adding synonyms, substituting preferred terms for non-preferred terms, or correcting errors). In other cases, however, creating the index could introduce keywords that the metadata contributor might consider misleading or inaccurate. For these reasons, when the keywords in a metadata record are “optimized” for use in the catalog index, a report should be sent to the contributor to solicit human feedback on a largely automated process: Are the corrections and additions to the catalog index valid? In the use case described below, this report to the metadata contributor is generated at specific *extension points* in the basic flow—specifically, when “an entry is made in the log.” Valid changes, as reported in this log, might prompt the metadata contributor to modify the authoritative copy of the metadata record (which might reside elsewhere). And, of course, there would be a feedback mechanism in place to undo those entries in the catalog index that are *not* valid.

Actors and SuD

List actors, people or things outside the system that either acts on the system (primary actors) or is acted on by the system (secondary actors). Primary actors are ones that invoke the use case and benefit from the result. Identify sensors, models, portals and relevant data resources. Identify the primary actor and briefly describe role.

- Catalog manager (primary actor, a machine) takes the actions to ingest metadata records and create a specialized index for the catalog.
- Metadata contributor (secondary actor, a human) evaluates validity of keyword matches

suggested by the catalog manager.
- Vocabulary services provided by a vocabulary server (not an actor, rather the system under development, or SuD).

Preconditions

Here we state any assumptions about the state of the system that must be met for the trigger (below) to initiate the use case. Any assumptions about other systems can also be stated here, for example, weather conditions. List all preconditions.

- Controlled vocabularies are available through the vocabulary services (SuD) in a form that the catalog manager can use.
- For each vocabulary in the server, the services provide the following information: preferred and alternative identifiers (names); keyword type (e.g., theme or place); and whether the vocabulary has transitive hierarchical relationships (i.e., if B is a subclass of A and C is a subclass of B, then C is a subclass of A).
- The catalog manager has mechanisms for filtering, as necessary, what it adds to the index: for instance, excluding “stop” words that are so general they have no value as search terms. Higher-level terms in the hierarchy (above the parent term) might also be excluded, even if the vocabulary is transitive.
- Controlled vocabularies provided will contain correct and quality-controlled values.
- Scope of vocabularies in the services is appropriate for the metadata collection.
- Metadata records being ingested by the catalog contain keywords, some (but not necessarily all) of which are from the vocabularies in the services.
- Metadata records being ingested conform to the FGDC CSDGM <http://www.fgdc.gov/metadata/csdgm/>.

Triggers

Here we describe in detail the event or events that brings about the execution of this use case. Triggers can be external, temporal, or internal. They can be single events or when a set of conditions are met, List all triggers and relationships.

The catalog ingest process is running and begins the step of processing the keywords in a particular metadata record. The catalog manager finds the Keywords elements in the metadata record.

Basic Flow

Often referred to as the primary scenario or course of events. In the basic flow we describe the flow that would be followed if the use case where to follow its main plot from start to end. Error states or alternate states that might be highlighted are not included here. This gives any browser of the document a quick view of how the system will work. Here the flow can be documented as a list, a conversation or as a story.(as much as required)

NOTE: The basic flow uses terminology specific to the FGDC CSDGM (e.g., “keyword thesaurus,” which may or may not be a thesaurus in the formal sense) but is also valid for other metadata standards.

1) Catalog manager finds the Keywords elements in the metadata record.

2) For each Keywords element, the catalog manager consults the vocabulary web services to see if the keyword thesaurus identified in the metadata record is one of the vocabularies

provided by the server, and if so, retrieves information about the vocabulary. If the keyword thesaurus is found in the server, proceed to step 3. (*Extension*: If the thesaurus identifier given in the metadata record is not the preferred identifier, the catalog manager makes an entry in the log.)

3) The catalog manager examines each keyword in turn and asks the vocabulary web services whether the text of the keyword matches a descriptor (preferred term) or a non-preferred term in the specified vocabulary. For each match the web services return the descriptor and other details needed by the catalog manager for step 4. (*Extension*: If the matching keyword is a non-preferred term in the specified vocabulary, the catalog manager makes an entry in the log.)

4) For each match the catalog manager creates an entry in the specialized index that includes the keyword type, descriptor, vocabulary identifier, and term identifier within the vocabulary and does the same thing for the parent term (and higher-level terms in the hierarchy, as warranted, if the vocabulary is transitive). Proceed to step 1 (for new vocabulary), step 3 (for new keyword), or step 8 (after all keywords have been examined).

5) If the text of the keyword does not match any descriptor or non-preferred term in the specified vocabulary, the catalog manager asks the vocabulary web services whether the text matches a term in any other vocabulary in the server. For each match the web services return the descriptor and other details needed by the catalog manager. The catalog manager creates an entry in the specialized index that includes the keyword type, descriptor, vocabulary identifier, and term identifier within the vocabulary and does the same thing for the parent term (and higher-level terms in the hierarchy, as warranted, if the vocabulary is transitive). (*Extension*: The catalog manager makes an entry in the log indicating that the term was not found in the specified vocabulary and which other vocabularies the term was found in.) Proceed to step 1 (for new vocabulary), step 3 (for new keyword), or step 8 (after all keywords have been examined).

6) If the declared keyword thesaurus is not recognized, then the catalog manager examines each keyword in turn and asks the web services whether the text of the keyword matches a descriptor or non-preferred term in any other vocabulary in the server. For each match the web services return the descriptor and other details needed by the catalog manager to create an entry in the index. (*Extension*: The catalog manager makes an entry in the log unless the declared keyword thesaurus is “none” or “general.”) Proceed to step 1 (for new vocabulary), step 3 (for new keyword), or step 8 (after all keywords have been examined).

7) If the declared keyword does not match any descriptor or non-preferred term in any vocabulary, then the catalog manager stores only the text of that term in the index with no additional information. (*Extension*: The catalog manager makes an entry in the log indicating that the term was not found in any vocabulary.) Proceed to step 1 (for new vocabulary), step 3 (for new keyword), or step 8 (after all keywords have been examined).

8) After all keywords in the metadata record have been examined, the use case ends. (*Extension*: The catalog manager sends a report to the metadata contributor based on log

entries, listing the errors, omissions, and irregularities that were identified and asking if the changes made to the specialized index are valid.)

Alternate Flow

Here we give any alternate flows that might occur. May include flows that involve error conditions. Or flows that fall outside of the basic flow.

2) If the keyword thesaurus is not found in the server, proceed to step 6.

3) If the keyword does not match any descriptor or non-preferred term in the specified vocabulary, proceed to step 5.

5) If the keyword does not match a descriptor or non-preferred term in any other vocabulary in the server, proceed to step 7.

6) If the keyword does not match a descriptor or non-preferred term in any vocabulary in the server, proceed to step 7.

Post Conditions

Here we give any conditions that will be true of the state of the system after the use case has been completed.

“Optimized” keywords from the metadata record have been added to the catalog’s specialized index, with additional information from the vocabulary server.

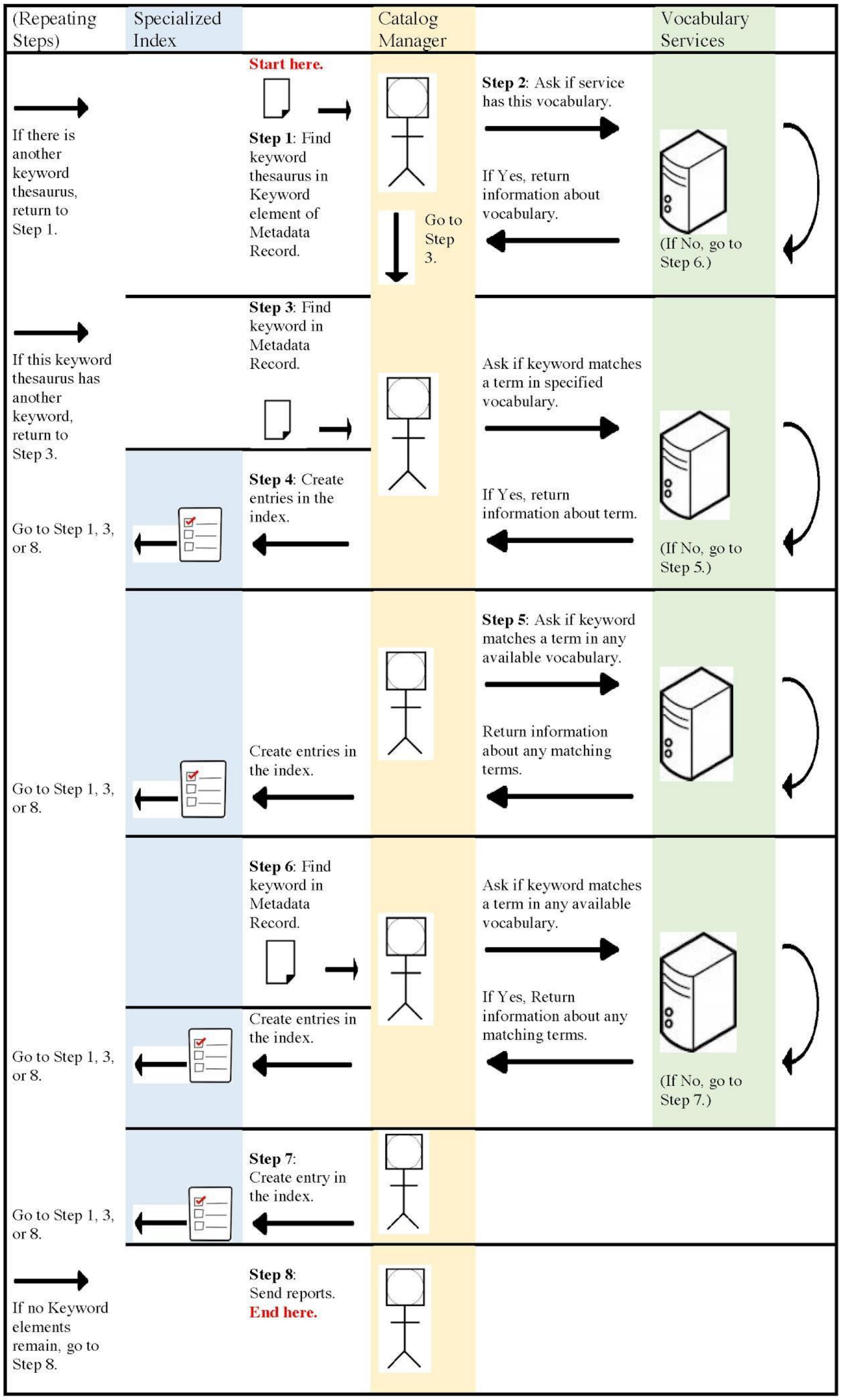
Based on the report sent by the catalog manager, the metadata contributor determines whether the changes made to the specialized index are valid, and if they are, might consider modifying the authoritative copy of the metadata record.

Activity Diagram

Here a diagram is given to show the flow of events that surrounds the use case. It might be that text is a more useful way of describing the use case. However often a picture speaks a 1000 words.

See following page.

Use Case 3: Create specialized indexes to enhance the searchability of metadata



(Repeating Steps)	Specialized Index		Catalog Manager		Vocabulary Services
If there is another keyword thesaurus, return to Step 1.		**Start here.** **Step 1**: Find keyword thesaurus in Keyword element of Metadata Record.	Go to Step 3.	**Step 2**: Ask if service has this vocabulary. If Yes, return information about vocabulary.	(If No, go to Step 6.)
If this keyword thesaurus has another keyword, return to Step 3. Go to Step 1, 3, or 8.		**Step 3**: Find keyword in Metadata Record. **Step 4**: Create entries in the index.		Ask if keyword matches a term in specified vocabulary. If Yes, return information about term.	(If No, go to Step 5.)
Go to Step 1, 3, or 8.		Create entries in the index.		**Step 5**: Ask if keyword matches a term in any available vocabulary. Return information about any matching terms.	
Go to Step 1, 3, or 8.		**Step 6**: Find keyword in Metadata Record. Create entries in the index.		Ask if keyword matches a term in any available vocabulary. If Yes, Return information about any matching terms.	(If No, go to Step 7.)
Go to Step 1, 3, or 8.		**Step 7**: Create entry in the index.			
If no Keyword elements remain, go to Step 8.		**Step 8**: Send reports. **End here.**			

Notes

There is always some piece of information that is required that has no other place to go. This is the place for that information.

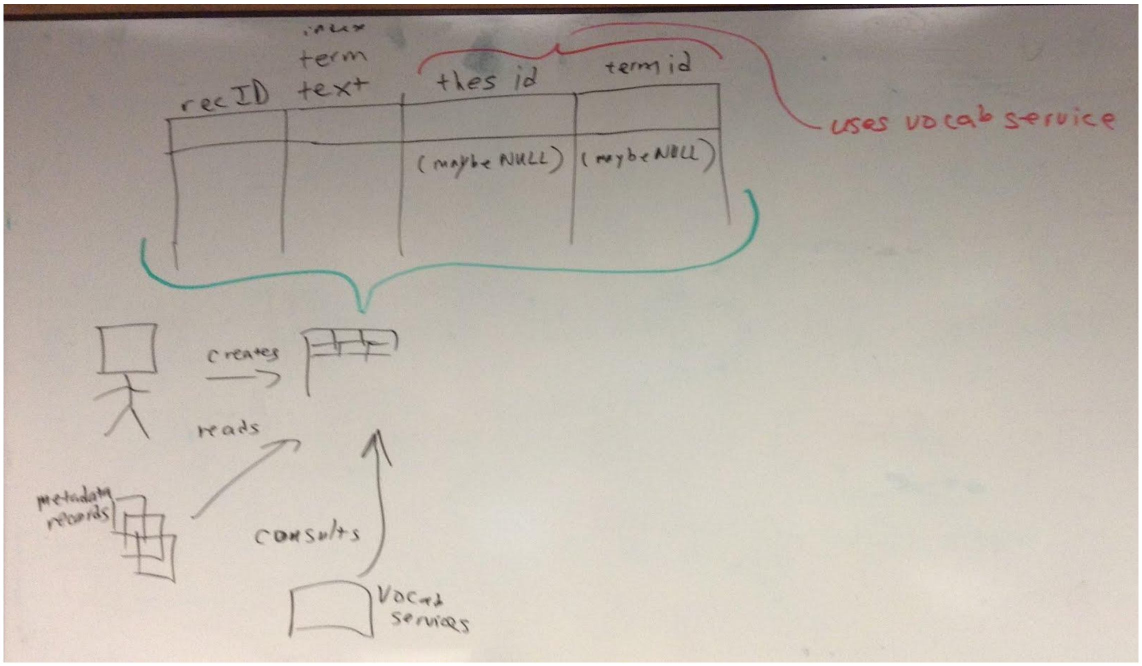
Basic flow assumes CSDGM metadata, but ISO, EML and other standards would also work in this use case.

In this use case the term "index" is used in the ordinary sense: "an indirect shortcut derived from, and pointing into, a greater volume of values, data, information or knowledge" (Wikipedia). Whereas a book index points to the specific pages where a subject is treated, the catalog index envisioned in this use case points to the specific metadata records where a subject is treated.

Vocabulary services: possible uses "behind the scenes" of a search system

How can the vocabulary services interact with the indexing process of metadata records? Are there ways in which term weighting can be influenced using controlled vocabulary terms? Can vocabulary services aid in the use of multiple vocabularies in an indexing process chain? Could there be a rule system (vocabulary A supersedes vocabulary B when the same term is found in both) to utilize both controlled vocabulary terms and uncontrolled terms as part of the indexing process? Can vocabulary type be added to the web services—the vocabulary of "stop" words?

Peter's diagram about using vocabulary services to populate a specialized index:



Example (above) illustrating a specialized keyword index in which metadata records are scanned to extract keywords. These keywords are mapped to controlled vocabularies and stored in a manner that would be easy to pass to the services to get additional information. In this example, it would also be possible to store in this specialized index the broader terms of the keywords, so that records indexed at the most specific level of detail can be returned in a search at a more general level that includes it.

If a declared keyword thesaurus is not provided by the server, a notification should be generated to the person(s) managing the vocabulary services system, who should evaluate whether that thesaurus has corresponding services and is suitable for inclusion in the vocabulary services system (either centralized or distributed).

If there are “X” number of instances of a keyword in the index that is not found in any of the vocabularies provided by the server, a notification should be generated for [the custodian of the USGS Thesaurus or some other controlled vocabulary] to review the term/concept for potential inclusion in [the USGS Thesaurus or some other controlled vocabulary].

This process could also be used to provide a list of possible keywords to the metadata collection managers in USGS programs. For example, if a thesaurus term appears in the title of a metadata record but does not appear as a keyword, this term could be suggested as an addition to the record. Likewise, if a keyword value is a non-preferred term, the preferred term could be suggested.

This requires knowing that the thesaurus is built with strictly “is a” relationships. Are BTs broader transitive? Are NTs narrower transitive? (“Transitivity” is an important metadata field for the thesaurus as a whole.)

Resources

In order to support the capabilities described in this Use Case, a set of resources must be available and/or configured. These resources include data and services, and the systems that offer them. This section will call out examples of these resources.

Other resources

Resource	Owner	Description	Availability	Source System
Overview of knowledge organization systems	CLIR	Systems of Knowledge Organization for Digital Libraries: Beyond Traditional Authority Files (Gail Hodge, 2000)	<http://www.clir.org/pubs/reports/pub91/pub91.pdf>	
Definition of precision and recall			<http://en.wikipedia.org/wiki/Precision_and_recall>	Wikipedia
ANSI/NIS	ANSI/NISO	Guidelines for the Construction, Format, and	<http://www.niso.org/standards/resour	

O Z39.19		Management of Monolingual Controlled Vocabularies	ces/Z39-19.html>	

Conceptual Models

The team developed two conceptual models (“concept maps”) for understanding the entities and relationships within the domain of the prototype vocabulary server. The simpler model (fig. 1) shows the fundamental concepts for using a vocabulary server in concert with a tool that creates formal metadata records (Use Case 1).

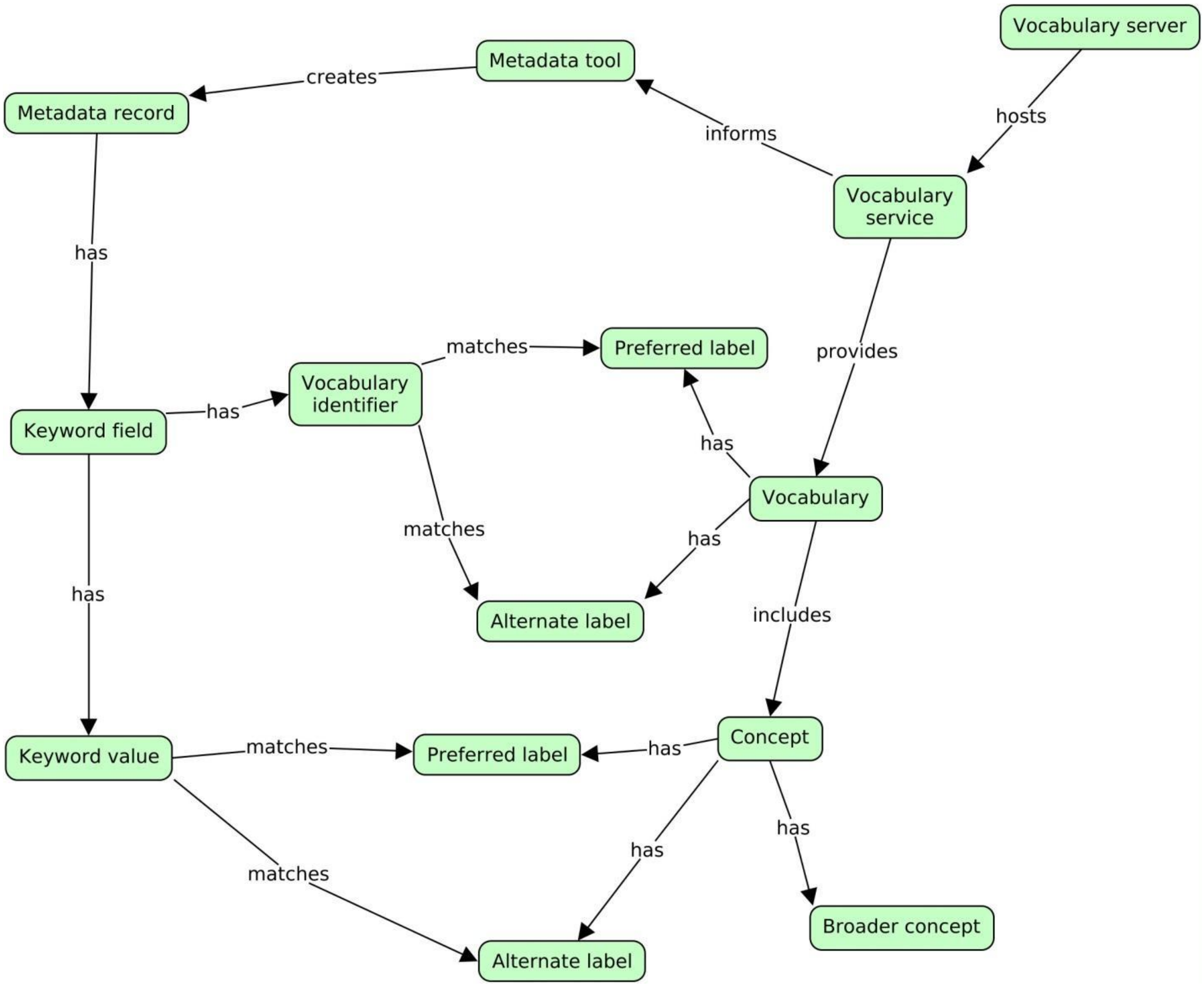


Figure 1: A conceptual model showing fundamental concepts for using a vocabulary server with a metadata tool, labeled with the terminology adopted by the use case team.

The more complicated model (fig. 2) is applicable to all three of the use cases; the green bubbles are the concepts that also appear in the simpler model above.

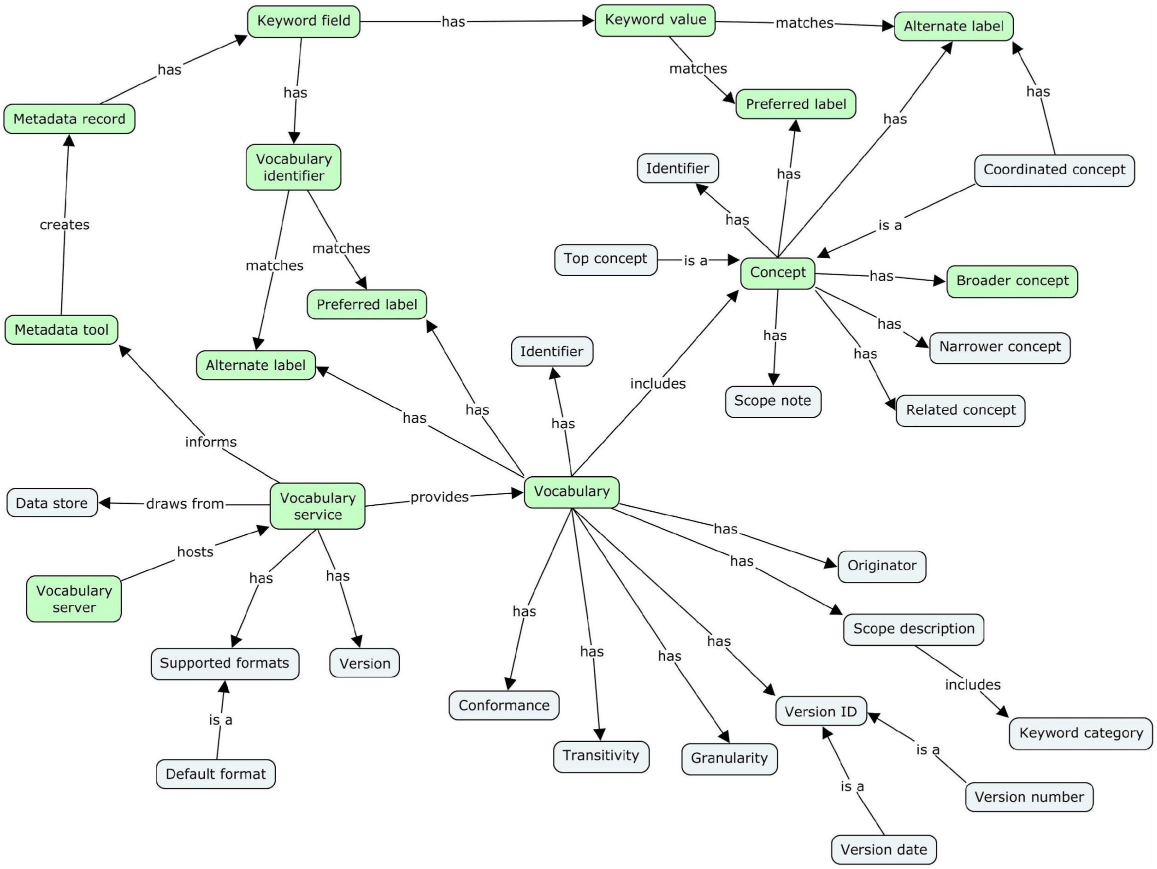


Figure 2: A more complicated conceptual model applicable to all three of the use cases; the green bubbles are the concepts that also appear in figure 1.

References

Bittner, Kurt, and Spence, Ian, 2003, Use case modeling: Boston, Addison-Wesley, 347 p.

Cockburn, Alistair, 2001, Writing effective use cases: Boston, Addison-Wesley, 270 p.

Fox, Peter, and McGuinness, D.L., 2008, TWC Semantic Web Methodology, <http://tw.rpi.edu/web/doc/TWC_SemanticWebMethodology>.

Lilly, Susan, 1999, Use case pitfalls—Top 10 problems from real projects using use cases, *in* Proceedings of the Technology of Object-Oriented Languages and Systems [TOOLS 99]: Washington, D.C., IEEE Computer Society, p. 174–183.

Schneider, Geri, and Winters, J.P., 2001, Applying use cases, second edition—A practical guide: Boston, Addison-Wesley, 245 p.